

## BAB III

### MEMBUAT STRUKTUR HALAMAN WEB

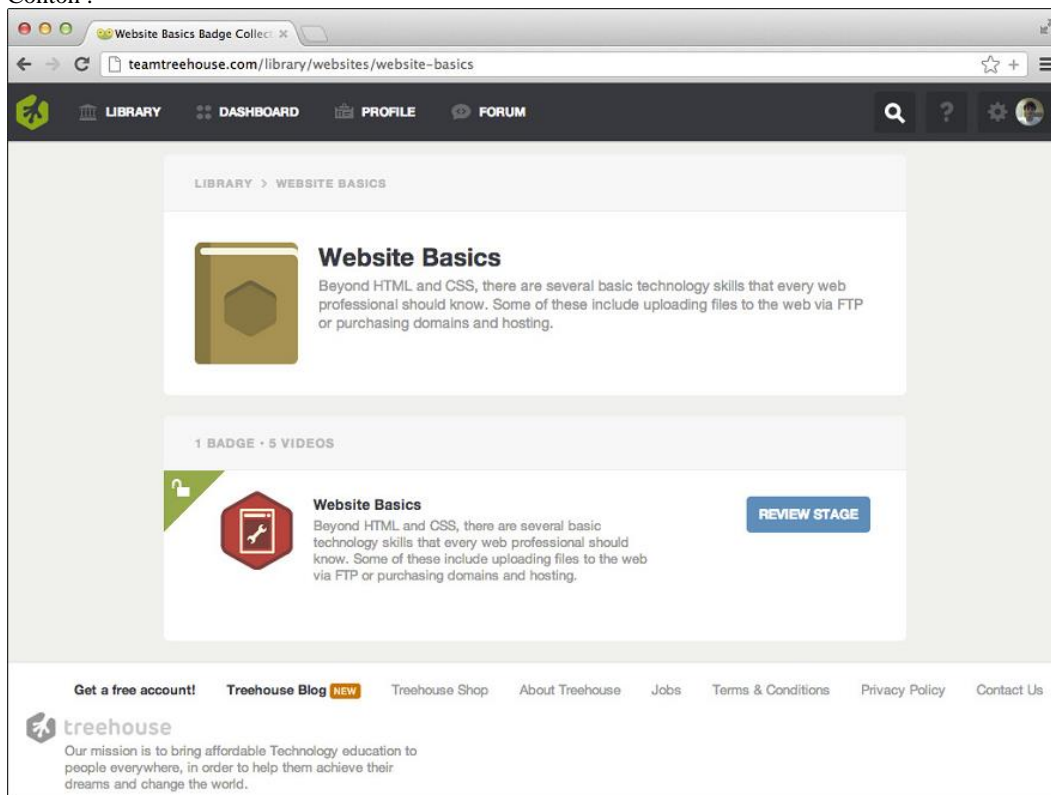
#### 3.1. Membuat Layout Template.

Pada bab ini, materi yang dibahas beserta contohnya akan diarahkan pada membangun website yang akan digunakan untuk mempromosikan sebuah restoran pizza Italia dengan tampilan yang stylish. Selain menu dan informasi tentang lokasi setiap restoran, juga akan tersedia berita-berita terbaru sehingga pelanggan dapat mengetahui semua produk-produk terbaik yang ada.

Langkah awal adalah membuat page layout (tata letak halaman), yang setidaknya memiliki struktur seperti berikut :

- Sebuah header
- Beberapa navigasi
- Beberapa isi halaman, dan
- Sebuah footer

Contoh :



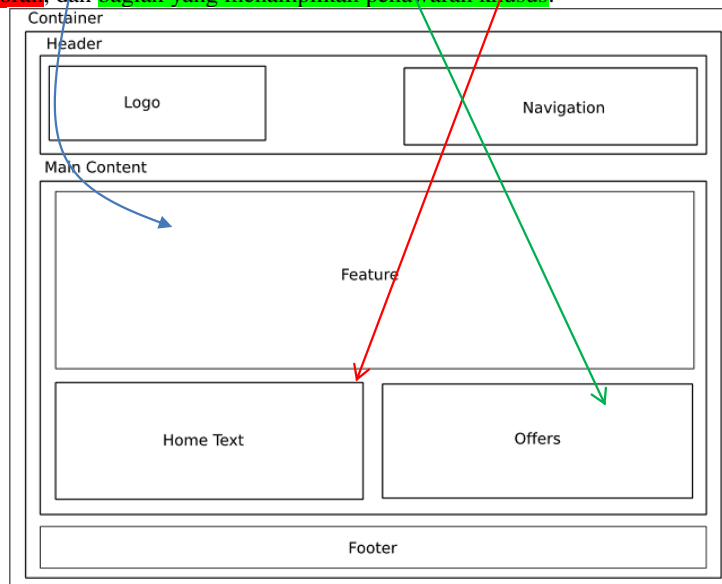
##### 3.1.1. Merencanakan Sitemap (peta situs)

Langkah pertama untuk membuat sebuah project website adalah **membuat *sitemap***, yang mendefinisikan **banyaknya halaman yang akan dimiliki situs** dan seperti apa halaman-halaman tersebut. Dari website restoran pizza Italia yang akan dibangun, maka kemungkinan sitemap nya adalah :

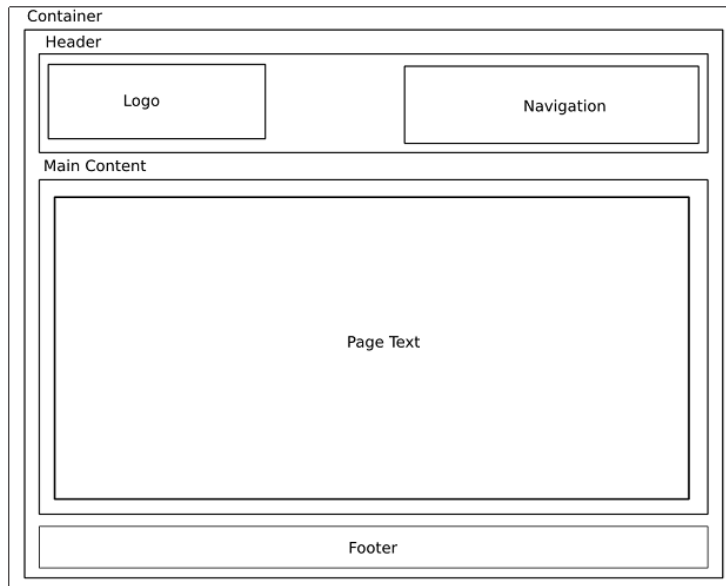
- a. *Home*
- b. *About*
- c. *News*
- d. *Menu*
- e. *Locations*
- f. *Sitemap*

### 3.1.2. Merencanakan Tata Letak Halaman (Page Layout)

Ini dilakukan untuk menentukan posisi halaman utama (home page): header, yang berisi logo dan navigasi; daerah content utama (main content); dan footer. Home page terdiri atas 3 bagian yaitu : fitur utama untuk menarik perhatian pengunjung, text box untuk informasi awal tentang restoran, dan bagian yang menampilkan penawaran khusus.



Untuk content page, dapat menggunakan layout yang mirip seperti ini, hanya dengan sedikit perubahan yaitu mengubah main content dari 3 bagian menjadi 1 bagian.



### 3.1.3. Membuat File-file Template

File template yang dibuat adalah **home page** dan **content page**.

#### a. Membuat template **home page**.

- Buat folder baru dengan nama **joes-pizza**.
- Ke dalam folder tersebut buat file **index.html**, masukkan/ketik kode berikut :

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Joe's Pizza Co. - New York's Best Pizza</title>
  </head>

  <body>

  </body>
</html>
```
- Simpan !

#### b. Membuat template **content page**.

- Buat file **about.html** dan letakkan di dalam folder **joes-pizza** !
- Masukkan kode berikut ke dalam **about.html** !

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>About - Joe's Pizza Co./title>
  </head>
  <body>
  </body>
</html>
```
- **Simpan !**

#### c. Menambahkan elemen **<div>** ke setiap template.

Comment [A1]: MM1,Senin27Okt2014

Elemen `<div>` digunakan sebagai container untuk mengelompokkan sejumlah elemen.

- Buka **index.html** dan **about.html** menggunakan text/web editor !
- Tambahkan kode berikut ke elemen `<body>` kedua file tsb!

```
<body>
<!-- Page Container -->
<div id="container">

</div>
</body>
```

- Simpan (kedua file tersebut) !

Syntax `<!--` dan `-->` digunakan untuk memberikan komentar, yang tidak akan ditampilkan oleh browser.

#### d. Menambahkan elemen `<header>` ke setiap template.

- Buka **index.html** dan **about.html** menggunakan text/web editor !
- Tambahkan elemen `<header>` ke kedua file tsb, tepatnya di dalam container !

```
<div id="container">
<!-- Header -->
<header>

</header>
</div>
```

- Simpan (kedua file tersebut)
- Buka kedua file tsb menggunakan web browser & periksa setiap halaman menggunakan developer tools untuk meyakinkan bahwa elemen `<header>` berada di dalam `<div>`.

#### e. Menambahkan logo di dalam elemen `<header>` ke setiap template.

- Buka **index.html** dan **about.html** menggunakan text/web editor !
- Tambahkan elemen `<div>` untuk logo ke kedua file tsb, tepatnya di dalam elemen `<header>`!

```
<div id="container">
<!-- Header -->
<header>
  <!-- Logo -->
  <div id="logo">Tempatkan logo di sini</div>
</header>
</div>
```

- Simpan (kedua file tersebut)

#### f. Menambahkan elemen `<nav>` ke setiap template.

- Buka **index.html** dan **about.html** menggunakan text/web editor !
- Tambahkan elemen `<nav>` di bawah logo, ke kedua file tsb !

```
<div id="container">
  <!-- Header -->
  <header>
    <!-- Logo -->
    <div id="logo">Tempatkan logo di sini</div>
    <!-- Navigation -->
    <nav>
      Navigation Links ditempatkan di sini
    </nav>
  </header>
</div>
```

- Simpan (kedua file tersebut)
- Buka kedua file tsb menggunakan web browser & periksa setiap halaman menggunakan developer toos (yang ada pada setiap browser) untuk meyakinkan bahwa elemen <nav> berada di bawah logo.

#### g. Menambahkan elemen <section>.

Elemen <section> digunakan untuk merepresentasikan bagian umum dari suatu halaman web. Misalnya untuk merepresentasikan bagian-bagian bab secara individu atau bagian yang berbeda dari suatu home page (seperti introduction, news, atau contact information). Contoh berikut menggunakan elemen <section> untuk mendefinisikan bagian-bagian main content (konten utama) yang berbeda.

- Buka **index.html** dan **about.html** menggunakan text/web editor !
- Tambahkan elemen <div> di bawah <header>!  

```
<div id="container">
  <!-- Header -->
  <header>...</header>
  <!-- Main Content -->
</div>
```
- Simpan file tersebut.
- Buka file-file tsb menggunakan web browser & periksa halaman menggunakan developer toos untuk meyakinkan bahwa elemen <div> terlihat dibawah <header>.

##### g.1. Menambah elemen <section> ke content page.

- Buka **about.html** menggunakan text/web editor !
- Tambahkan elemen <section> dengan nilai atribut **id = page-text**, diletakkan di antara  

```
<div id="container">
  <!-- Header -->
  <header>...</header>
  <!-- Main Content -->
  <div>
    <!-- Page Text -->
    <section id="page-text">
      Tempatkan produk terbaik di sini !
    </section>
  </div>
</div>
```
- Simpan file !
- Buka about.html menggunakan web browser & periksa setiap halaman menggunakan developer toos untuk meyakinkan bahwa elemen <section> terlihat diantara main content <div>.

##### g.2. Menambah elemen <section> ke home page.

Tata letak (layout) home page sedikit lebih kompleks karena memiliki 3 bagian content terpisah : satu untuk fitur banner utama, satu untuk informasi perusahaan, dan satu untuk special offers. Elemen <section> digunakan dalam setiap bagian ini.

- Buka **index.html** menggunakan text/web editor !
- Tambahkan elemen <section> di dalam main content <div> dan berikan **id = feature**.  

```
<!-- Main Content -->
<div>
  <!-- Feature -->
  <section id="feature">
    Fitur banner utama di sini.
```

- ```

    </section>
</div>

```
- Tambahkan elemen <section>berikut dibawahnya dan berikan **id = home-text**.

```

<!-- Main Content -->
<div>
    <!-- Feature -->
    <section id="feature">...</section>

    <!-- Home Text -->
    <section id="home-text">
        Informasi tentang perusahaan di sini
    </section>
</div>

```
  - Tambahkan elemen <section>lagi dibawahnya di dalam main content <div> dan berikan **id = offers**.

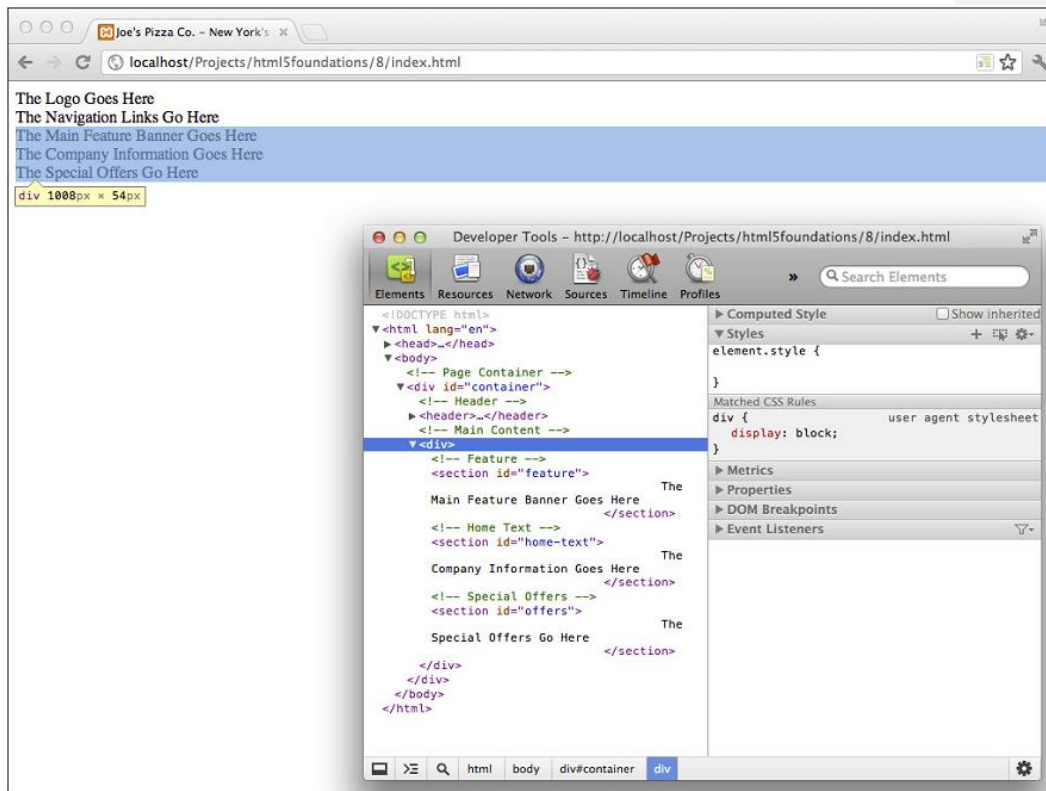
```

<!-- Main Content -->
<div>
    <!-- Feature -->
    <section id="feature">...</section>

    <!-- Home Text -->
    <section id="home-text">...</section>

    <!-- Special Offers -->
    <section id="offers">
        Penawaran khusus di sini
    </section>
</div>

```
  - Simpan file !
  - Buka index.html menggunakan web browser & periksa setiap halaman menggunakan developer toos untuk meyakinkan bahwa elemen <section> ditampilkan terurut dari fitur, home text, kemudian special offers.



Dengan memberikan **id** kepada setiap elemen akan memudahkan saat menggunakan CSS untuk memberikan style pada halaman web.

**h. Menambahkan elemen <footer> ke setiap template.**

Elemen ini biasanya berisi copyright notices (catatan hak cipta) dan link yang ada di bagian bawah sebuah halaman web. Seperti pada <header> dan <nav>, lebih dari satu <footer> dapat ditambahkan pada sebuah halaman.

- Buka [index.html](#) dan [about.html](#) menggunakan text/web editor !
- Tambahkan elemen <footer> dibawah main content <div>, pada kedua file tsb !

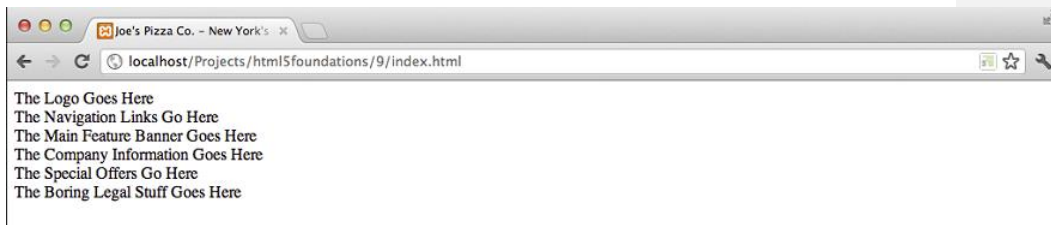
```
<div id="container">
  <!-- Header -->
  <header>...</header>

  <!-- Main Content -->
  <div>...</div>

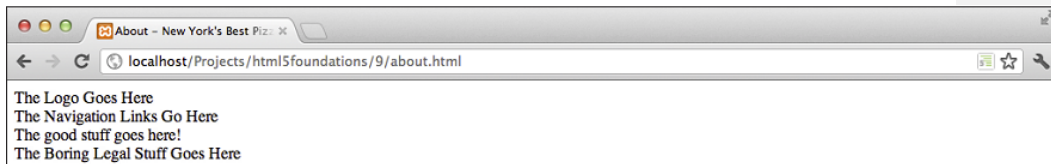
  <!-- Footer -->
  <footer>
    The Boring Legal Stuff Goes Here
  </footer>
</div>
```

- Simpan (kedua file tersebut)

Baiklah, sekarang anda telah membuat main layout untuk template home maupun content page.



Gambar home page template



Gambar content page template.

### 3.2. Menggunakan *Cascading Style Sheets* (CSS).

CSS adalah bahasa yang digunakan untuk mengontrol aspek visual dari halaman web, baik posisi elemen, latar belakang warna dan image/gambar, border, font, & aspek design lain. CSS.

#### 3.2.1. Menghubungkan CSS ke HTML

Untuk menghubungkan CSS ke HTML, gunakan elemen `<link>`. Contoh : `<link rel="stylesheet" href="style.css">`. Ada 2 atribut : rel dan href. Atribut rel digunakan untuk menggambarkan hubungan dari file yang berhubungan. Atribut href digunakan untuk menunjuk lokasi file stylesheet.

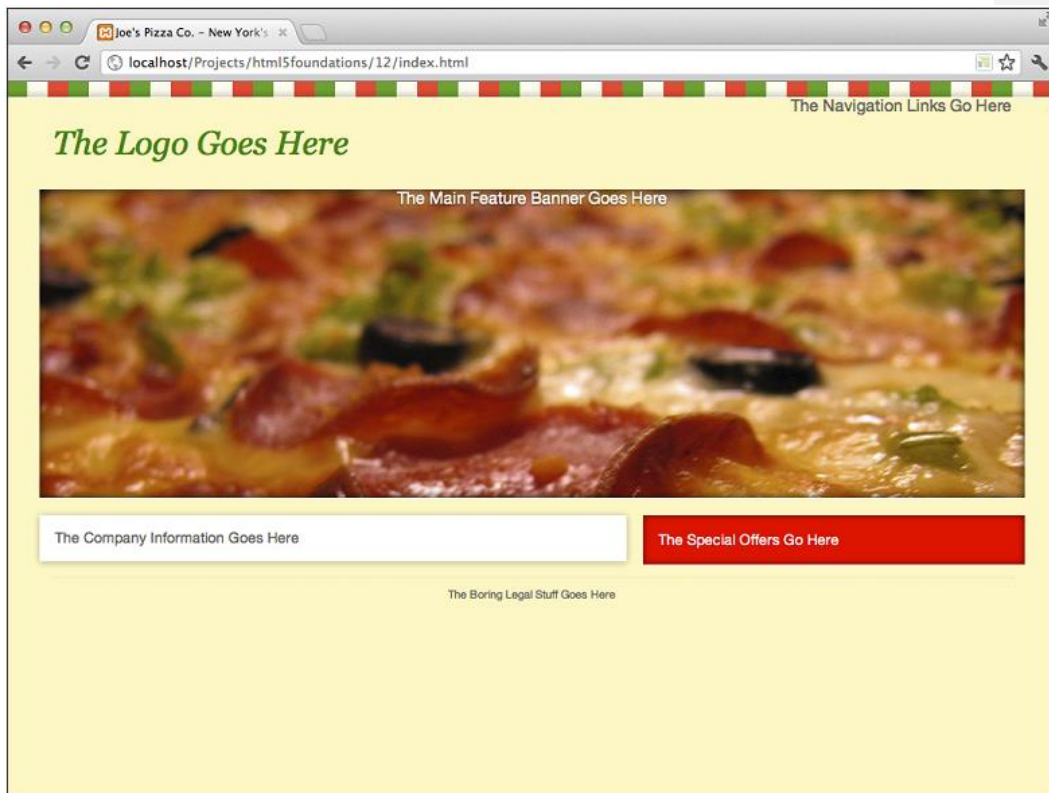
##### **Latihan :**

- Buat folder baru dengan nama **css**. Tempatkan file .css yang diberikan guru ke dalam folder tsb !
- Buat folder baru dengan nama **img**. Tempatkan file-file image (bodyBg.png dan featureBg.png) yang diberikan guru ke dalam folder tsb !
- Buka [index.html](#) dan [about.html](#) menggunakan text/web editor !
- Tambahkan elemen `<link>` di dalam elemen `<head>` !
 

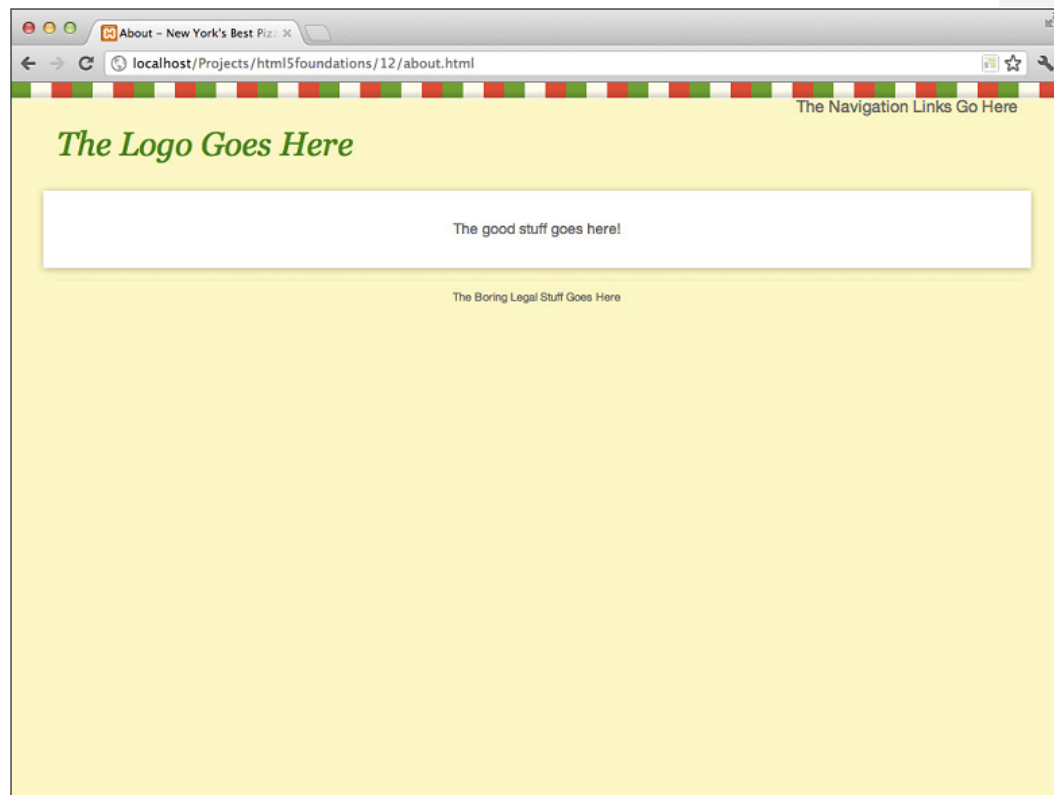
```
<head>
  <meta charset="utf-8">
  <title>...</title>

  <link rel="stylesheet" href="css/style.css">
  ...
</head>
```
- Simpan (kedua file tersebut)
- Buka kedua template menggunakan browser dan lihat hasilnya.





**FIGURE 2-13** The home page template, with CSS.



**FIGURE 2-14** The content page template, with CSS.